

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-345164

(43)Date of publication of application : 14.12.1999

(51)Int.Cl.

G06F 12/00

G06F 13/00

H04M 11/00

(21)Application number : 10-154824

(71)Applicant : SONY CORP

(22)Date of filing : 03.06.1998

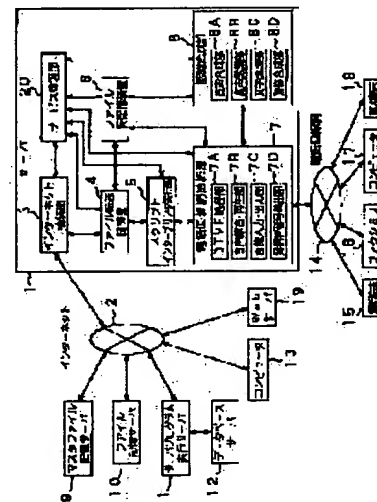
(72)Inventor : TSUNODA HIROSHI  
FUJIMURA SATOSHI  
KANDA SATORU  
ONOKI WATARU

## (54) INFORMATION PROCESSOR

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To set the function of cache to be used at obtaining of information through the Internet according to a request from a user by individually controlling the storage state of information obtained and stored from a network.

**SOLUTION:** When a service user sent connects through a telephone line network 14 with a server 1 by using a telephone terminal 15, the server 1 transmits a guidance voice to a telephone terminal 15. When the service user inputs a desired service code from the operator of a telephone terminal 15, the server 1 receives and converts the signal by a telephone line network connecting section 7. When this service code is proper, a service managing part 20 extracts a script corresponding to the service code from a file temporary storage device 6, transfers it to a script interpreter executing section 5, and interprets and executes response script. Thus, plural services can be realized by one server 1 by assigning a code number to each response script.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

BEST AVAILABLE COPY

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-345164

(43) 公開日 平成11年(1999)12月14日

(51) Int.Cl.<sup>6</sup>

識別記号

F I

G 0 6 F 12/00

5 4 6

G 0 6 F 12/00

5 4 6 M

13/00

3 5 4

13/00

3 5 4 A

H 0 4 M 11/00

3 0 3

H 0 4 M 11/00

3 0 3

審査請求 未請求 請求項の数7 O L (全 23 頁)

(21) 出願番号

特願平10-154824

(22) 出願日

平成10年(1998)6月3日

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 角田 弘史

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72) 発明者 藤村 聡

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72) 発明者 神田 悟

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(74) 代理人 弁理士 小池 晃 (外2名)

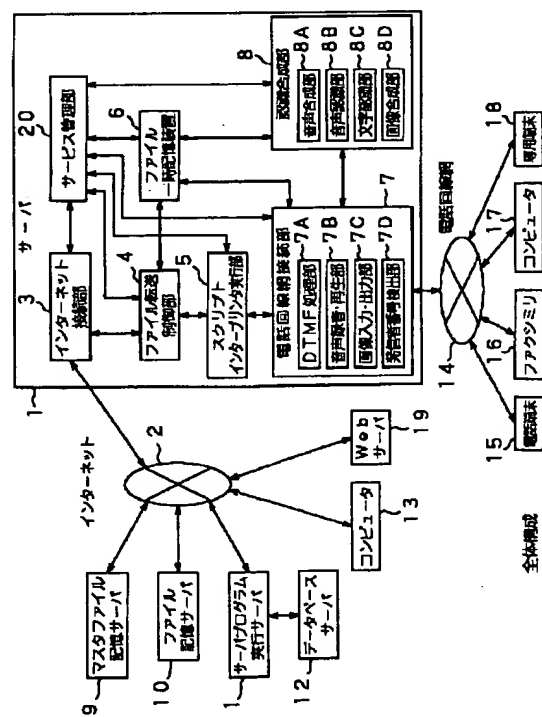
最終頁に続く

(54) 【発明の名称】 情報処理装置

(57) 【要約】

【課題】 インターネット等を介して情報を取得する場合に使用されるキャッシュの機能をユーザに要求に応じて設定可能にする。

【解決手段】 インターネット2に接続されるインターネット接続部3と、少なくともインターネット2を介して取得した情報を記憶するファイル一時記憶装置6と、当該ファイル一時記憶装置6のキャッシュ機能を個別に制御可能にするためのスクリプトインタープリタ実行部5及びファイル転送制御部4とを有する。



**【特許請求の範囲】**

**【請求項1】** コンピュータネットワークに接続されるネットワーク接続手段と、

少なくとも上記コンピュータネットワークを介して取得した情報を記憶する記憶手段と、

上記コンピュータネットワークを介して取得して上記記憶手段に記憶した情報の、当該記憶手段上での記憶状態を個別に制御可能な制御手段と、を有することを特徴とする情報処理装置。

**【請求項2】** 実行手順ファイルを解釈実行する解釈実行手段を備え、

上記コンピュータネットワーク上の複数のサーバのうちの何れかより上記実行手順ファイルを選択的に取得することを特徴とする請求項1記載の情報処理装置。

**【請求項3】** 上記制御手段は、上記実行手順ファイルの実行手順に基づいて動作することを特徴とする請求項2記載の情報処理装置。

**【請求項4】** 上記実行手順ファイルにおける実行手順の制限、変更を個別に制御するためのファイルを取得することを特徴とする請求項2記載の情報処理装置。

**【請求項5】** 上記解釈実行手段は、上記実行手順ファイルの種類に応じて、実行手順を実行する処理プログラムを選択することを特徴とする請求項2記載の情報処理装置。

**【請求項6】** 上記実行手順ファイルと共に上記処理プログラムを上記複数のサーバのうちの何れかから取得することを特徴とする請求項5記載の情報処理装置。

**【請求項7】** 上記実行手順ファイルにリンクするファイルを、予め上記複数のサーバのうちの何れかから取得して上記記憶手段に保存しておくことを特徴とする請求項2記載の情報処理装置。

**【発明の詳細な説明】****【0001】**

**【発明の属する技術分野】** 本発明は、例えばインターネット等を用いて情報を取得する情報処理装置に関する。

**【0002】**

**【従来の技術】** 以前は、一般の人々が不特定多数の人に対して情報を発信し、流通させていくことは困難であったが、近年はインターネットが急速に普及し、その上でいわゆるWWW (World Wide Web) や電子メールを利用することにより、情報を発信することが比較的容易になってきた。特に、WWWについては、情報を受け取る人にとって読み易いよう、情報をレイアウトするソフトウェアが普及してきており、多くの人々が様々な情報 (例えば自分の研究成果、プロフィール、趣味の情報など) を公開している。

**【0003】** ここで、WWWや電子メールといった情報をネットワーク上から取り出して閲覧するためには、現在のところパーソナルコンピュータ (いわゆるパソコン、以下、適宜PCとする) や専用の端末をネットワー

クに接続する方法が一般的である。

**【0004】**

**【発明が解決しようとする課題】** ところで、従来より、パーソナルコンピュータを用い、インターネットを介して情報を取得する場合には、いわゆるWWWブラウザとよばれる専用のソフトウェアが使用されている。このWWWブラウザにおいては、その機能の一つとして、インターネット上のサーバから取得した情報を、ハードディスク上に設けられたキャッシュメモリ領域に保存しておき、後に再び同じ情報を取得しようとした場合に、インターネットを介して再度取り出すようなことはせずに、このハードディスクのキャッシュメモリ領域に保存しておいた情報をあたかもインターネットを介して取得したかのようにして使用する機能が存在する。

**【0005】** このキャッシュ機能において、キャッシュとして保存する情報は、インターネットを介して新たな情報を取得した場合にのみ保存され、既にキャッシュメモリ領域に保存されている情報と同じデータについては保存しないようになされている。また、キャッシュ機能では、インターネットを介して取得した情報を、ユーザが設定した任意の期間だけ一律に保存するようになされている。したがって、上記設定された一定の期間が過ぎると、何れのデータであっても一律にキャッシュメモリ領域から自動的に消去されてしまうことになる。

**【0006】** しかし、インターネットを介して取得した情報のなかには、一律にキャッシュさせずに、キャッシュしなくてよい情報やできるだけ長くキャッシュしておきたい情報があり、また、特定の時期までキャッシュして欲しい情報などを区別したい場合がある。さらに、インターネット上のサイト側で情報が更新されているような場合、その更新をチェックできればより望ましい。

**【0007】** そこで、本発明はこのような状況に鑑みてなされたものであり、インターネット等を介して情報を取得する場合に使用されるキャッシュの機能を、ユーザに要求に応じて設定可能にする情報処理装置を提供することを目的とする。

**【0008】**

**【課題を解決するための手段】** 本発明の情報処理装置においては、コンピュータネットワークに接続されるネットワーク接続手段と、少なくともコンピュータネットワークを介して取得した情報を記憶する記憶手段と、コンピュータネットワークを介して取得して記憶した情報の、当該記憶手段上での記憶状態を個別に制御可能な制御手段とを有することにより、上述した課題を解決する。

**【0009】**

**【発明の実施の形態】** 本発明の好ましい実施の形態について、図面を参照しながら説明する。

**【0010】** 本実施の形態では、例えばインターネットや公衆電話回線網等を用いた一般の人々による情報の発

信及び取得を容易に実現可能にするシステムに本発明の情報処理装置を適用した例に挙げている。すなわち、WWWや電子メールといった情報をネットワーク上から取り出して閲覧するためには、現在のところパーソナルコンピュータ（いわゆるパソコン、以下、適宜PCとする）や専用の端末をネットワークに接続する方法が一般的であり、それにそれら情報を閲覧するためのソフトウェアの操作に習熟していることが必要であるが、本実施の形態では、パーソナルコンピュータや専用の端末を必要とせず、また、閲覧用のソフトウェアの操作に習熟する必要もなく、さらに、利用料金が必要なインターネット接続業者（いわゆるインターネットプロバイダ）を介さずとも、それらの情報を取得できるようなシステムを例に挙げている。また、公衆電話回線網を用いたサービス提供例としては、例えば従来より電話応答サービスやファクシミリサービス等のシステムが存在するが、このシステムでは、応答サービスを提供したい者（個人或いは企業等の団体）が例えば自らがデータを作成し、この応答サービス専用の装置を用意して運営したり、または専門の業者にデータの作成と装置の運用を依頼するようになされており、前者においては装置を導入するコストが大きく、後者においては頻繁なデータの更新が難しく、依頼のコストも発生し、何れにしてもWWWのように個人が気軽に情報を発信するという目的には不向きであり、さらに、同システムにおいてはインターネット上に数多く存在する種々の情報（WWWなど）を利用することができず、多くはサービスを運営する個人あるいは団体等が自ら所有している情報にアクセスできないのに対し、本実施の形態では、情報の発信や情報の取得を容易且つ低コストにて実現するようなシステムを例に挙げている。

【0011】以下、

## 1 概要

## 2 システム構成

### 2.1 全体構成

### 2.2 ファイル

#### 2.2.1 応答手順スクリプト

#### 2.2.2 データ

### 2.3 応答サービスコード

### 2.4 基本動作

#### 2.4.1 スクリプトとデータの作成

#### 2.4.2 応答スクリプトの登録

#### 2.4.3 スクリプトの利用

### 2.5 拡張機能

#### 2.5.1 音声合成

#### 2.5.2 音声入力

#### 2.5.3 外部ファイルの取り込み

#### 2.5.4 ファイル転送制御部

#### 2.5.5 ファイルのキャッシュ

#### 2.5.6 ファイルの先読み

#### 2.5.7 スクリプトの機能制限・機能変更

#### 2.5.8 スクリプトエンジンの自動選択

#### 2.5.9 スクリプト実行関連ファイルの自動ダウンロード

#### 2.5.10 スクリプト記述外ユーザ制御

#### 2.5.11 接続先端末アドレス/種別によるスクリプトの自動選択

#### 2.5.12 サーバプログラム、データベースアクセス

### 2.6 ダウンロードファイルのキャッシュの詳細

#### 2.6.1 キャッシュイン手順

#### 2.6.2 キャッシュアウト手順

### 2.7 キャッシュ制御の拡張の詳細

#### 2.7.1 個別キャッシュインチェック

#### 2.7.2 個別キャッシュアウトチェック

### 2.8 ダウンロードファイルの先読みの詳細

の順に説明を行う。

## 【0012】1 概要

本発明の情報処理装置にかかるシステムにおいては、WWWと同様に、インターネットを用いて情報を管理することにより、従来の電話応答サービスやファクシミリサービス等よりも気軽に情報を作成・管理できる環境を提供すると共に、情報を受け取る手段として電話端末やファクシミリ端末を利用できるようにすることにより、より多くの人々が気軽に情報を受け取ることができるようにしている。

【0013】図1には、本発明の情報処理装置が適用される代表的な利用形態として、パーソナルコンピュータ（PC30）でサービスデータを作成・管理し、電話端末15にてサービスの提供を受けるような実施の形態のシステム構成を示す。つまり、本実施の形態のシステムは、前述したWWWと電話応答サービス等の双方の長所を兼ね備え、短所を補うようにしたシステムである。

【0014】ここで、図1に示す本実施の形態のシステムが、従来より存在する電話応答システムと大きく異なる点は、応答サービスを運用する装置（サーバ1）がインターネット2に接続されており、当該インターネット2上に接続された他の端末（PC30等）から応答サービスとしてのデータをダウンロードできるということである。

【0015】すなわち、本実施の形態のシステムによれば、応答サービスのアプリケーション作成側（PC30）はサービスの運営に携わらなくても良く、自己の端末（PC30）をサービス運営のために使用する必要がない。これは、現在のWWWにおける利点の一つでもある。したがって、従来の電話応答サービスに比べ、より多くの人々が応答サービスアプリケーションの作成に気軽に参加することができるようになり、WWWに加えてさらに多くの便利な情報が一般の人々によりもたらされることになると期待される。

【0016】さらに、応答サービスを運用する装置（サ

サーバ1)がインターネット2に接続されていることを利用し、インターネット2上に存在する既存のWWWの情報をも利用することができるようになされている。これにより、WWWの膨大な既存データを活用した音声応答サービスを提供できるようになる。

【0017】一方、本実施の形態のシステムが従来のWWWのシステムと大きく異なる点は、サーバ1が電話回線網14に繋がれており、一般の電話端末15から接続してサービスを受けられるという点である。これにより、既に世の中に広く普及している電話端末15を用いてサーバ1上の情報にアクセスすることができるようになる。すなわち、従来のWWWのシステムでは、インタ

```
<HTML>
<TITLE>test page</TITLE>
<BODY bgcolor=white>
  <H1>Test Page</H1>
  <IMG src=ball.gif>This is a test page.<BR>
  For more information, go <A href=detail.html>here</A>.
</BODY></HTML>
```

これに対し、本実施の形態のシステムにおいて、電話端末15の利用者に提供する情報を記述するためのプログラミング言語は、電話で情報を聴覚的な形式でやり取りすることを想定し、音声を再生する順番やタイミング、電話端末使用者からのボタン入力の扱いなどをプログラミングできるようになされている。なお、本実施の形態のシステムでは、後述するように電話端末以外の端末（ファクシミリ端末など）からの利用をも想定しているため、それらの端末からも利用可能となるようにプログラミング言語を設計している。

【0020】したがって、従来のWWWシステムでは「HTMLで記述された文書ファイルとそれに付随する画像・音声等のデータ」を蓄積しているが、本実施の形態のシステムでは「専用言語で記述されたファイルとそれに付随する画像・音声等のデータ」を使用する。以下の説明では、この「専用言語で記述されたファイル」を「応答スクリプトファイル」または単に「応答スクリプト」と呼ぶことにする。

【0021】上記応答スクリプトで記述された応答サービスの簡単な例は、いわゆるテレホンサービスである。すなわち、そのサービスの提供者に電話をかけると特定の案内音声再生され、利用者に情報の選択を求め、利用者が所望の情報を示すコード番号を電話端末の操作子により入力すると、対応する情報の音声を再生するというものである。このように本実施の形態のシステムでは、電話端末からアクセスしてくる利用者に対しては従来の電話応答サービスと同様のサービスを提供することが可能ようになされており、さらに後述するように従来のサービスには無いサービスをも提供することが可能である。

## 【0022】2 システム構成

インターネット2に接続された端末（PC）からサーバにアクセスし、情報を視覚的な形で閲覧することが一般的であったが、本実施の形態のシステムによれば、電話端末15からアクセスし、聴覚的な形で情報を引き出すことができる。

【0018】上述したようなシステムを実現するため、本実施の形態では、専用のプログラミング言語を定義する。なお、従来のWWWでは情報を記述するために、下記のようないわゆるHTML（Hyper Text Markup Language）と呼ばれる言語を使用している。当該HTMLは情報を視覚的に閲覧することを前提としている。

【0019】

## 2.1 全体構成

図2には、本発明の実施の形態のシステム全体構成と、当該システムの情報ネットワークに接続されるサーバ1の内部構成を示す。

【0023】この図2において、サーバ1は、図1の対話プログラム実行部31を構成するサービス管理部20、インターネット接続部3、ファイル転送制御部4、スクリプトインタプリタ実行部5、電話回線網接続部7、認識合成部8と、図1の記憶装置32の一部を構成するファイル一時記憶装置6とを有してなる。このサーバ1は、電話回線網14を介し、電話端末15、ファクシミリ端末16、コンピュータ端末17、専用端末18等のユーザ端末と通信し、また必要に応じてインターネット2を介して他のサーバ（後述するマスタファイル記憶サーバ9、ファイル記憶サーバ10、サーバプログラム実行サーバ11、データベースサーバ12、webサーバ19等）やコンピュータ（例えばパーソナルコンピュータ、以下、PCとする）13と通信する。これにより、このサーバ1は、ユーザ端末からの要求に応じて、種々のサービスを提供する。

【0024】すなわち、このサーバ1において、電話回線網接続部7は、電話回線網14を通じて音声データ、ファクシミリデータ、DTMF（Dual Tone Multi-Frequency）信号等を送受信し、音声録音・再生部7B、画像入力・出力部7C、DTMF処理部7Aにて、それぞれのデータに対し、エンコード、デコード、誤り訂正等の処理を行う。また、発信者番号検出部7Dにおいては、このサーバ1に接続したユーザ端末の電話番号を検知可能であれば検出する。このように、サーバ1では、電話端末15、ファクシミリ端末16における操作子の操作や入力される音声、ファクシミリデータを検出して

種々のサービスを提供できるようになされている。なお、電話回線網14に接続するユーザ端末は、上記電話端末15やファクシミリ端末16に限らず、コンピュータ端末17や専用端末18、携帯電話、いわゆるPHS(personal handyphone system)等であっても良い。

【0025】ファイル転送制御部4は、インターネット接続部3からインターネット2を介して得られる音声データ、ファクシミリ端末のイメージデータ、WWW(World Wide Web)の文書データ、応答手順スクリプトデータ等をサーバ1の内部におけるデータ形式に適宜変換してスクリプトインタプリタ実行部5やファイルー時記憶装置6に出力し、またこれとは逆にスクリプトインタプリタ実行部5やファイルー時記憶装置6から送られてきたデータを適宜適切な形式に変換してインターネット2に出力する。ただし、ここでインターネット2におけるデータ形式とサーバ1の内部におけるデータ形式は、必ずしも異なっている必要はない。

【0026】上記インターネット2に接続されているマスタファイル記憶サーバ9は、サーバ1の働きを制御するためのマスタファイルを保持している。

【0027】ファイル記憶サーバ10は、応答手順スクリプトデータや音声、画像等のデータ(総称してリモートファイルと呼ぶ)を保持している。

【0028】データベースサーバ12は、内部にデータベースを保持しており、サーバプログラム実行サーバ11からの要求により、データベースの内容を適宜サーバプログラム実行サーバ11に出力し、また逆にサーバプログラム実行サーバ11から入力されたデータにしたがってデータベースを更新する。

【0029】サーバプログラム実行サーバ11は、インターネット2に接続された他の端末からの要求と内部に保持しているサーバプログラムの内容にしたがい、適宜データベースサーバ12と通信してインターネット2と情報のやり取りを行う。

【0030】PC13は、インターネット2に接続されており、適宜サーバ1やマスタファイル記憶サーバ9、ファイル記憶サーバ10、サーバプログラム実行サーバ11と通信する。これにより、PC13の利用者はこれら各種サーバ上に蓄積された情報を管理することができるようになされている。ただし、セキュリティの観点から、各サーバにおいてはサーバ外部からの管理操作において充分な認証動作を行う。

【0031】このような仕組みにより、サーバ1は、インターネット2を介し、マスタファイル記憶サーバ9からマスタファイルを取り出し、ファイル記憶サーバ10との間でリモートファイルをやり取りし、サーバプログラム実行サーバ11を介してデータベースサーバ12のデータをやり取りするようになされている。さらに、サーバ1は、インターネット2に接続された他のPC13と通信し、PC13に情報を送付したり、逆にPC13

からの命令に従ってサーバ1の動作を変更したりすることができるようになされている。

【0032】サーバ1のスクリプトインタプリタ実行部5は、インターネット2を介して入力された、あるいはファイルー時記憶装置6に保持されたマスタスクリプトの内容にしたがって、適宜動作を切り替え、ファイル転送制御部4やファイルー時記憶装置6、電話回線網接続部7を制御し、これらの装置とデータのやり取りを行う。

【0033】ファイルー時記憶装置6は、ファイル転送制御部4、認識合成部8と通信し、それらから入力されたデータを内部に保持し、これらの装置からの制御にしたがって保持したデータを適宜出力あるいは更新あるいは消去する。

【0034】ファイルー時記憶装置6は、少なくとも一つのスクリプト管理ファイルを持つ。スクリプト管理ファイルは、例えば図3に示すような構造をしており、応答手順スクリプトファイル名と応答サービスコードの対応を管理する。

【0035】認識合成部8は、電話回線網接続部7の制御に従い、電話回線網接続部7あるいはファイルー時記憶装置6から入力されたデータに対し音声合成、音声認識、文字認識、画像合成等の処理を行い、その結果を電話回線網接続部7あるいはファイルー時記憶装置6に出力する。この処理のために、認識合成部8は内部に音声合成部8A、音声認識部8B、文字認識部8C、画像合成部8Dを内蔵しており、それらを適宜用いる。

## 【0036】2.2 ファイル

### 2.2.1 応答手順スクリプト

本実施の形態では、電話端末15等の端末を使用するユーザが電話回線網14を通じてサーバ1と行う対話を司るプログラムを応答手順スクリプト(適宜、応答スクリプト、スクリプトなどとする)と呼んでいる。そして、計算機上のデータとしてこのプログラムがファイル化されたものを、応答手順スクリプトファイル(適宜、応答スクリプトファイル、スクリプトファイルなどとする)と呼んでいる。

【0037】スクリプトファイルは、サーバ1のスクリプトインタプリタ実行部5で解釈され、その内容にしたがってサーバ1が動作する。すなわち、当該スクリプトファイルは、サーバ1の動作を制御するフローや命令を記したものである。以下に、応答スクリプトの例を示す。

### 【0038】

```

{
    while (TRUE) {
        &wait_call;
        &off_hook;
        &play_sound('hello.snd');
        &on_hook;
    }
}

```

また、図4にはその応答スクリプトの実行手順を示す。この図4のフローチャートにおいて、ステップSP1では利用者の端末から電話がかかってくるのを待ち、ステップSP2では利用者から電話がかかってくる時に回線が接続される。ステップSP2にて回線が接続されると、ステップSP3では、ファイル名が「hello.snd」となされている音声ファイルを再生する。その後は、ステップSP4にて回線を切断する。

【0039】上記応答手順スクリプトの中で、特にサーバ1の起動時に最初に読み込まれるものをマスタスクリプトと呼ぶ。

【0040】スクリプトは電子的にファイル化されたものがファイルー時記憶装置6かマスタファイル記憶サーバ9かファイル記憶サーバ10に存在しなければならない。

#### 【0041】2.2.2 データ

前述のスクリプト例のプログラム中の「hello.snd」のように、スクリプトファイルからは他のデータファイルを参照することができるようになされている。この例では音声データを参照しているが、他にもテキストデータや画像データ等を利用することが可能である。さらに、同じ音声データについても、様々なフォーマットのデータファイルを利用することが可能である。それらのデータは電子的にファイル化されたものであり、ファイルー時記憶装置6や、ファイル記憶サーバ10等に格納される。

#### 【0042】2.3 応答サービスコード

本実施の形態のシステムにおいて、電話端末15等により電話回線網14から接続してくるユーザに対して行われる、応答手順スクリプトによって構成されるサービスには、それぞれ固有のコード番号を割り当てることとし、これを応答サービスコードと呼ぶ。これにより、天気予報サービス、道路交通情報サービス、地図印刷サービスなどサーバ1に登録された様々なサービスを、ユーザはユーザ端末にて当該コード番号により選択することができる。

【0043】前記図3に示した管理ファイルは、このコード番号と実際の応答サービスを表すスクリプトとを関連付けるものであり、当該サービスにおいてユーザがユーザ端末から当該コード番号を入力すると、サーバ1はこの管理ファイルに基づいて対応するスクリプトを決定することができるようになされている。

【0044】応答サービスコードは、応答手順スクリプトの作者がサーバ1にスクリプトを登録する際にサービス管理部19により生成される。

#### 【0045】2.4 基本動作

本発明実施の形態によるシステムの基本動作として、以下の項目が挙げられる。

##### 【0046】

- ・ 応答手順スクリプト及び関連データの作成
- ・ サーバ1への同スクリプトの登録
- ・ 同スクリプトによるサービスの利用
- ・ 同スクリプト及び関連データのメンテナンス

以下では、上記各項目のそれぞれについて説明する。

##### 【0047】2.4.1 スクリプトとデータの作成

前記2.2節で述べた通り、本実施の形態のシステムによって実現され、電話回線網14に接続した機器を利用するユーザに供されるサービスは、応答手順スクリプトとそれに関連するデータにその内容が記述される。

【0048】応答手順スクリプトは、前記2.2.1節で例示したテキストファイルであるが、その言語仕様は本発明に含まない。言語仕様にしたがってさえいれば誰でも作成することが可能である。応答手順スクリプトに関連するデータ（音声、画像など）も誰でも作成することが可能である。スクリプト、データ共に作成後にサーバ1に登録されて始めて、電話回線網14を通じて利用されることができる。

##### 【0049】2.4.2 応答スクリプトの登録

インターネット2に接続されたPC13からサーバ1に応答スクリプトを登録する方法は、様々な例を考えることができる。

【0050】図5には、PC13からインターネット2を経由してサーバ1に応答スクリプトを登録するにおいて、サーバ1の動作手順の一例を示す。

【0051】この図5に示すフローチャートにおいて、ユーザが例えばPC13上でいわゆるweb閲覧ソフトを起動し、スクリプトの登録のためにサーバ1にPC13を接続させると、webサーバ19が働き、当該webサーバ19では、ステップSP11にてスクリプト登録用のwebページをPC13のweb閲覧ソフトに送信する。上記PC13上で実行したweb閲覧ソフトによりディスプレイ上に表示される応答スクリプト登録画面としては、図6のような画面を一例として挙げるができる。この応答スクリプト登録画面上には、名前の入力部とファイル名を入力部が表示され、さらにその入力の登録を行うか否かのソフトキー表示（「OK」、「RESET」）等が表示される。

【0052】応答スクリプトの作成者は、PC13において応答スクリプトを作成し、この画面を開いて、名前を入力部とファイル名を入力部の空欄に、自分の名前と応答スクリプトのファイル名を入力する。これにより、PC13からは、当該PC13上にある応答スクリプト

そのもの、または応答スクリプトの所在を表すアドレスがHTTP (Hyper Text Transfer Protocol) によってサーバ1へ送られる。そして、サーバ1は、ステップSP12の処理に移り、送られてきたデータを受信する。

【0053】続いて、サーバ1は、ステップSP13の処理に移り、スクリプト自身が送られてきたのか、スクリプトのアドレスが送られてきたのかを判定し、アドレスが送られてきたのであればステップSP14の処理に移ってそのアドレスからスクリプト自身を取り出す。なお、アドレスとしては、例えばいわゆるURL (Unified Resource Locator) を使用することができ、その場合、サーバ1は当該URLに記されたプロトコルにしたがってインターネット接続部3よりインターネット2を通じてPC13からスクリプトを取り出す。

【0054】このようにして何れかの方法でスクリプトを受信したサーバ1は、次にステップSP15の処理に移り、スクリプトの内容をチェックし、文法的な誤りがないか判定する。ここで誤りが検出された場合は、ステップSP16の処理に移り、PC13に対してエラーメッセージを送信する。これにより、PC13のweb閲覧ソフトの画面上にはエラーメッセージが表示され、ユーザに対して修正の後、改めて登録するよう促す。

【0055】ステップSP15のチェックで誤りが検出されなかった場合、サーバ1はステップSP17の処理に進み、サービス管理部20がこのスクリプトをファイル時記憶装置6に保存する。さらに、ステップSP19の処理に進み、サービス管理部20はスクリプトに対応するサービスコードを生成し、webサーバ19に知らせる。すると続くステップSP20においてwebサーバ19は通知されたサービスコードをユーザに知らせるためのwebページを生成し、そのデータをPC13に送信する。これにより、ユーザはPC13上のweb閲覧ソフトに画面上において、サーバ1により発行されたサービスコードを確認することができる。図7には、このときweb閲覧ソフトの画面上に表示される画面出力の一例を示す。すなわちこの図7の例では、登録結果として、「あなたの応答スクリプトファイルは正しく登録されました。この応答スクリプトのアクセス番号は3487です。」等の表示がなされる。

#### 【0056】2.4.3 スクリプトの利用

図8には、応答スクリプトによるサービスをサービス利用者が受ける場合の、サーバ1の動作手順を示す。

【0057】この図8において、サービス利用者が電話端末15を用い、電話回線網14を通じてサーバ1に接続すると、サーバ1はステップSP31の処理に移り、電話端末15に対し、ガイダンス音声 (例えば「ご希望のサービスコードを入力してください」など) を送信する。これに対し、サービス利用者が希望のサービスコード (図7の例で登録されたスクリプトに対するものであれば3487) を電話端末15の操作子により入力する

と、サーバ1はステップSP32の処理に移り、その信号を電話回線網接続部7で受信・変換し、内部形式にする。

【0058】続いてステップSP33で、この値をサービス管理部20において適正かどうか判定し、適正でなければ、すなわち入力されたサービスコードに対応するサービスが提供可能でなければ、ステップSP36の処理に移って電話回線網接続部7から電話回線網14を経由してサービス利用者の電話端末15に対してサービスコードが不正であることの音声による通知を行い、続いてステップSP32の処理に戻って再度サービスコードが入力されるのを待ち受ける。

【0059】ステップSP33の判定においてサービスコードが適正であれば、ステップSP34の処理に移って、サービス管理部20がそのサービスコードに対応するスクリプトをファイル時記憶装置6から取り出し、スクリプトインタプリタ実行部5に渡す。

【0060】続いてステップSP35に移り、スクリプトインタプリタ実行部5が渡された応答スクリプトを解釈・実行する。

【0061】このように個々の応答スクリプトにコード番号を割り付けることにより、一台のサーバ1で複数のサービスを実現することができる。図9には1台のサーバ1で複数のサービスを実現する場合の構成例を示す。

【0062】この図9において、インターネット2上には、例えば「天気予報」の応答スクリプトS1の作者のPC41と、例えば「ニュース」の応答スクリプトS2の作者のPC42がそれぞれ接続されており、両者はそれぞれサーバ1に接続してスクリプトを登録する。これにより、サーバ1は、既に登録されている例えば「交通情報」の応答スクリプトS3と合わせてそれらのスクリプトを登録し、それぞれのスクリプトにコード番号を発行する。電話端末15を用いてサーバ1に接続した利用者は、それぞれのコード番号 (この例ではS1かS2かS3のスクリプト) のうち所望のサービスを提供するスクリプトの番号を入力することにより、所望のサービスを受けることができる。

#### 【0063】2.5 拡張機能

##### 2.5.1 音声合成

前述のように、サーバ1は、予め用意された音声データを利用者の電話端末15等へ送信することが可能であるが、テキスト音声合成を用いることにより、テキストデータから音声を生成して送信することも可能である。

【0064】図10には、前記図2に示した全体構成の中で、音声合成に関係する部分のみを抜き出して表す。

【0065】この図10において、ファイル時記憶装置6はプログラムと音声データのみならず、テキストデータをも保持することができるようになされている。

【0066】そしてスクリプトインタプリタ実行部5は、読み込んだプログラムに従い、適宜テキストデータ

をファイル転送制御部4に要求する。ファイル転送制御部4は、要求されたファイルをファイル一時記憶装置6から取り出して電話回線網接続部7に送信し、当該電話回線網接続部7が同データを認識合成部8へ送る。

【0067】認識合成部8では、音声合成部8Aにてこのデータを音声に変換し、生成された音声データを電話回線網接続部7に送信する。そして電話回線網接続部7より電話回線14を通じてユーザの電話端末15に音声を送信される。

【0068】なお、ファイル一時記憶装置6に格納され

```
{
    while(TRUE) {
        &wait_call;
        &off_hook;
        &play_text("ただいまサービスを停止しております。");
        &play_text("明日以降おかけ直し下さい。");
        &on_hook;
    }
}
```

このように本実施の形態のシステムでは、テキストデータから音声を生成して対話サービスに使用することができるため、対話サービスを作成する者が必ずしも音声データを用意する必要がなく、より簡単にサービスを作成することが可能となる。

#### 【0071】2.5.2 音声入力

前述のように、サーバ1はユーザからの入力として電話端末15の操作子によるDTMF信号を受け取り、解釈することが可能であるが、さらにユーザによる電話端末15からの音声入力を受け取るようにもなされている。ユーザが電話端末15のマイクロホンから入力した音声は、電話回線網14を通じて電話回線網接続部7に入力される。実際にこの音声データの処理はスクリプトインタプリタ実行部5が、実行中のプログラムにしたがって決定し、電話回線網接続部7を制御することにより行う。

【0072】例えば、プログラム中で電話端末15からの音声をファイル一時記憶装置6に録音するよう指定されている場合は、電話回線網接続部7に入力された音声データは音声録音・再生部7Bにて内部形式に適宜変換され、ファイル一時記憶装置6に保存される。

【0073】また、プログラム中で電話端末15からの音声を音声認識に使用するよう指定されている場合は、電話回線網接続部7に入力された音声データは認識合成部8に送られ、音声認識部8Bにおいて音声認識処理にかけられ、その結果がスクリプトインタプリタ実行部5に送られる。

【0074】このように、ユーザにより電話端末15か

```
{
    while (TRUE) {
        &wait_call;
```

たテキストデータを認識合成部8に送る場合においては、前述のような方法に限らず、ファイル一時記憶装置6より直接認識合成部8に送るようになされていても良い。その場合は、対象となるデータのファイルの識別子をスクリプトインタプリタ実行部5から電話回線網接続部7を通じて認識合成部8に通知する。

【0069】以下に、特定の文章を読み上げるスクリプトの一例を示す。

【0070】

ら入力された音声入力をサーバ1が受け取れるようになっていることにより、電話端末15を介したユーザの音声を録音して後に再生するサービスや、複数メニューからの選択を電話端末15の操作子によらず音声で行うことができるサービスなどを実現するプログラムを扱うことが可能となる。

#### 【0075】2.5.3 外部ファイルの取り込み

前述の図2に示したシステム例では、応答スクリプトやそれに付随するデータをサーバ1内のファイル一時記憶装置6に収めているが、本発明ではこれらスクリプトやデータをサーバ1の外に置くことも可能である。

【0076】前記2.2.1節で取り上げた応答スクリプトでは、スクリプトの中で音声ファイルを再生するようになされている（前記応答スクリプトの例）。このスクリプトにおいて音声ファイル名を指定している部分をより一般化することにより、スクリプトファイルが保存されているのと同じ場所のみならず、ネットワーク上の様々な場所にあるデータを指定することができる。

【0077】インターネット上のデータの所在（以下、住所と呼ぶ）を示す方法として代表的なものにURL（Unified Resource Locator）があるが、本実施の形態のシステムでもURL又はそれと同様の機能を持つものを用いる。

【0078】例えば、前述の2.2.1節で取り上げた応答スクリプトの例に対し、下記のスクリプトでは、音声ファイル名を指定する部分でURLを指定している。

【0079】

```

&off_hook;
&play_sound('http://www.foo.co.jp/data/hello.snd');
&on_hook;
}
}

```

このスクリプトがサーバ1に登録され、ファイル時記憶装置6に格納された状態を図11に示す。

【0080】この図11において、データを指定したスクリプトをサーバ1で実行する場合、サーバ1は、必要に応じて適宜指定されたデータファイルを、インターネット接続部3からインターネット2を通して取り出す。例えば、スクリプトインタプリタ実行部5がスクリプトを解釈・実行しているときにデータファイルの取り出しをインターネット接続部3に指示し、インターネット2を通じてファイル記憶サーバ10から取り出された音声データを、音声録音・再生部7Bで再生するなどという形が考えられる。

【0081】さらに、図12に示すように、スクリプト自体をサーバ1の外部に置くことも可能である。

【0082】この図12の例の場合、応答スクリプトの作成者は、スクリプト自体を登録する代わりに、スクリプトの所在を表すURLをサーバ1に登録する。サーバ1はファイル時記憶装置6内のスクリプト一覧に登録されたURLを保存し、応答サービスの中で必要となった時点でインターネット2を通じて登録された応答スクリプトの実体をファイル記憶サーバ10から取り出し、実行する。なお、図12では、スクリプトとデータファイルが同一のサーバ（ファイル記憶サーバ10）上に置かれているが、それぞれ異なるサーバ上に置かれていても良い。この場合、それぞれにURLが割り当てられており、それにしたがってサーバ1から取り出すことが可能であれば良いのである。また、URLの代わりに同様の機能（ネットワーク上のデータの所在を一意に決定する機能）を有する符号化法を用いても良い。

【0083】上記図11や図12のように、スクリプトやスクリプトから使用されるデータをサーバ1の外部に置くことにより、それらの変更や追加を容易に行うことができるようになる。例えば、スクリプトの作成者が自分のwebサーバを管理している場合、そのサーバにスクリプトを置くことにより、随時スクリプトの内容を変更することができる。

#### 【0084】2.5.4 ファイル転送制御部

本実施の形態のシステムのサーバ1では、図13に示すように、内部にファイル転送制御部4を置くことにより、より柔軟なデータ転送を実現することができる。なお、図13は、サーバ1内のファイル転送制御部4とそれに関連する部分のみを抜き出して示している。

【0085】この図13において、ファイル転送制御部4は、インターネット接続部3を介して得られる応答スクリプトやそれに付随する各種のデータをサーバ1の内

部におけるデータ形式に適宜変換してスクリプトインタプリタ実行部5やファイル時記憶装置6に出力し、またこれとは逆にスクリプトインタプリタ実行部5やファイル時記憶装置6から送られてきたデータを適宜適切な形式に変換してインターネット2に出力する。ただし、ここでインターネットにおけるデータ形式とサーバ1の内部におけるデータ形式は、必ずしも異なっている必要はない。

#### 【0086】2.5.5 ファイルのキャッシュ

ファイル転送制御部4はインタプリタ実行部5から要求があったデータをインターネット2上の他のサーバから取り出した際に、そのデータをファイル時記憶装置6に保存し、後に再び同じデータの要求があったときに、インターネット2上のサイトから再度取り出さずにファイル時記憶装置6に保存してあるデータを返すようになされていてもよい。これによって同じファイルの読み込みにかかる時間のために応答スクリプトの実行が遅延することを防ぐことができる。このダウンロードファイルのキャッシュに関しては、後述する2.6節の「ダウンロードファイルのキャッシュの詳細」にて詳しく説明する。

【0087】なお、ダウンロードファイルのキャッシュにおいて、スクリプト作成者の意志によっては一律にキャッシュさせずに、キャッシュしなくてよいファイルやできるだけ長くキャッシュして欲しいファイル、特定の時期までキャッシュして欲しいファイルなどの区別を行いたい場合がある。また、キャッシュされているファイルがダウンロード後にサイト側で更新されているかのチェックもする必要があるが、その更新チェック頻度も指定したい場合がある。このような要求にはサイト側に管理ファイルを持っていて、それをダウンロードすることによって対応することができる。このキャッシュの拡張制御に関しては、後述する2.7節の「キャッシュ制御の拡張の詳細」にて詳しく説明する。

#### 【0088】2.5.6 ファイルの先読み

本実施の形態のシステムにおいては、実行中あるいは実行前の応答スクリプトで必要となるファイルを予めファイル時記憶装置6に読み込んでおくようになされてもよい。これによってスクリプト実行中に、あるファイルが必要になった時点で初めて読み込みを始めるのに比べて、応答スクリプトの実行が遅延することを防ぐことができる。実行中の応答スクリプトで必要となるファイルの先読みに関しては、後述する2.8節の「ダウンロードファイルの先読みの詳細」にて詳しく説明する。

【0089】なお、ダウンロードファイルの先読みにお

いて、スクリプト作成者の意志によっては、ファイルにより他のものよりも早く先読みしたほうがよいものや、また先読みしたくないものがあつたりすることがある。

【0090】そのような場合は、スクリプト自身に明示的に特定ファイルの先読みを優先指示するコマンドを記述したり、先読みを禁止するコマンドを記述したりすることによって、そのような要求にも応えることが可能になる。

【0091】2.5.7 スクリプトの機能制限・機能変更  
本実施の形態のシステムでは、様々なサイトからの応答スクリプトファイルをダウンロードして実行することが可能であるが、どの応答スクリプトも無条件にあらゆる機能を実行できるようにすると、逆にシステムのセキュリティやリソース量などで問題になる場合がある。

【0092】このような場合、サイト毎の、又はファイル別のスクリプトの実行権限を設定したファイルを用意しておくことにより、その設定に応じてスクリプトに記述されたコマンドを機能制限したり、または機能変更したりすることによってセキュリティやリソース量などの問題を解消することが可能になる。

【0093】リソース量使用量などの制限では、例えば録音最大時間やダウンロード可能なファイルのサイズ、同時にキャッシュされるファイルの最大サイズやキャッシュアウトされる優先度、応答スクリプトの呼び出し可能レベルなどの設定項目が考えられる。また、セキュリティによる制限では、ファイル一時記憶装置6内のファイルのアップロード可否やシステムファイルに対するアクセスの可否などの設定項目が考えられる。

【0094】なお、機能変更とは、例えばアクセス制限された音声ファイルを再生しようとしたときに、エラー音声のファイルを再生するなどの代替処理を行うことである。これも機能制限と一緒に、代替処理を設定しておくことにより可能になる。

【0095】2.5.8 スクリプトエンジンの自動選択  
本実施の形態のシステムでは、応答スクリプトをスクリプトインタプリタ実行部5で解釈／実行する。この解釈／実行するプログラム（スクリプトエンジン）は、必ずしも1つである必然性はない。応答スクリプトの記述言語が複数あっても、それぞれに対してスクリプトエンジンを複数用意しておけば、応答スクリプトを実行する前にスクリプトインタプリタ実行部5がどのスクリプトエンジンで実行できる種類かを自動的に判別して該当するスクリプトエンジンに解釈／実行させることができる。

【0096】種類の判別には、例えばスクリプト自身の最初にその記述言語の名称を記述しておいたり、対応するスクリプトエンジンの名前を記述しておいたりすることによって対応できる。

【0097】また、応答スクリプトを記述する言語が同じであっても、言語ソースの形でそのまま記述されたも

のを実行するだけでなく、予め中間言語の形に変換された応答スクリプトを実行したい場合もある。このような場合に対しても、ソースおよび中間言語それぞれに応じたスクリプトエンジンを用意しておくことにより対応することができる。

【0098】2.5.9 スクリプト実行関連ファイルの自動ダウンロード

スクリプトエンジンは、必ずしも予めサーバ1上に用意されている必要はなく、必要に応じてマスタファイル記憶サーバ9からダウンロードして、ファイル一時記憶装置6に格納するようになされていてもよい。例えばそのスクリプトエンジンで解釈／実行される必要がある応答スクリプトを実行しようとしたときにスクリプトインタプリタ実行部5がファイル転送制御部4に要求して必要なスクリプトエンジンをダウンロードしたり、または、ファイル転送制御部4が応答スクリプトをダウンロードしたときに、それを解釈／実行するのに必要なスクリプトエンジンがファイル一時記憶装置6に無かった場合に自分でダウンロードするというようになされていてもよい。

【0099】また、自動でダウンロードするファイルは、スクリプトエンジンでなくともよく、例えば応答スクリプトを解釈／実行しようとしたときに、そのスクリプト内で使われている特定のコマンドを実行するために必要なプログラムやデータがサーバ1上に無かった場合に、必要なファイルをマスタファイル記憶サーバ9からダウンロードしてくるようになされていてもよい。

【0100】こうすることにより、サーバ1上に予め考え得るあらゆるプログラムやデータを予め用意しておく必要がなく、ファイル一時記憶装置6のリソースの節約になる。また、サーバ1の運用後にスクリプトエンジンやデータなどが更新されたり新しく作られたとしても、サーバ1の運用を停止することなく必要なときに最新のファイルをダウンロードすることができるので、常にシステムを最新の状態に保つことができる。また、それらのファイルをファイル一時記憶装置6に置くことにより、他のダウンロードファイルと同様のキャッシュイン、キャッシュアウト、ファイルの先読みの機能を使うことが出来、必要なくなったファイルが何時でもシステムのリソースを無駄使いすることがなくなる。

【0101】2.5.10 スクリプト記述外ユーザ制御  
PCなどで利用されるいわゆるWWWブラウザにおいては、あるホームページのHTMLファイルを読み込んでHTMLの記述に従って表示したりするだけではなく、転送中や表示中においてもユーザの意図に従ってファイルの読み込み途中で中断したり、直前に表示したページを再度表示させたりということができるようになっている。画面を使った端末の場合はその機能のためのソフトウェアを画面上に設け、それをユーザが好きなときに押すことによって可能になる。しかし、電話端末のよう

に、ユーザが入力できるインターフェースが極めて限定されている端末などの場合は、応答スクリプト上でそのインターフェースによる入力を待っていたりして、システムが前記のような特別な用途に占有できないことが多い。

【0102】このような場合は、スクリプトインタプリタ実行部5または電話回線網接続部7において、そのインターフェースから特定のシーケンスの入力があった場合に、応答スクリプト自身にその入力を渡さずに、スクリプトインタプリタ実行部5が独自の解釈を行って制御するようになされればよい。

【0103】このようにすれば、電話端末15の操作子から、例えば連続して「\*\*\*\*1」が入力された場合は、直前のスクリプトに戻るようにし、また例えば連続して「\*\*\*\*0」が入力された場合は最初のスクリプトに戻るなどと制御することが応答スクリプト自身の記述に関係なく可能になる。もちろん、操作子による操作のみならず、音声による例えば「ストップ」や「戻る」等の入力によって同様の制御を行うようにすることも可能である。

【0104】2.5.11 接続先端末アドレス／種別によるスクリプトの自動選択

ところで、本実施の形態のシステムでは、当該システムを通信端末で利用する利用者が当該ネットワークに接続するときに、同じスクリプトのサービスを利用することが便利であるとは限らない。

【0105】そこで例えば、電話回線網制御部7に発信

```
{
    while (TRUE) {
        &wait_call;
        &off_hook;
        &read_aloud_html('http://www.foo.co.jp/data/news.html');
        &on_hook;
    }
}
```

また、読み上げに限らず、WWW上のデータやサーバプログラム（CGIなど）から得られる情報を様々な利用することができるようになされている。例えば、あるサーバプログラムがそのときの現在時刻を返すようになさ

```
{
    while (TRUE) {
        &wait_call;
        &off_hook;
        $time = &get_web('http://www.foo.co.jp/cgi-bin/time.cgi');
        &play_text("ただいまの時刻は $time です。");
        &on_hook;
    }
}
```

このようなスクリプトを処理する場合、スクリプトインタプリタ実行部5がこのスクリプトを解釈し、サーバ

番号検出部7Dを備え、ユーザの通信端末からの接続時にその番号をスクリプトインタプリタ実行部5に伝えることにより、スクリプトインタプリタ実行部5がその番号に応じた応答スクリプトを選択して実行するようになされていてもよい。

【0106】こうすることにより、例えば予め登録したユーザに応じたサービスをすぐに提供したりすることができる。

【0107】また、端末の種類を検知できる機能を電話回線網接続部7に備え、ユーザの通信端末からの接続時にその端末の種類をスクリプトインタプリタ実行部5に伝えることにより、スクリプトインタプリタ実行部5がその番号に応じた応答スクリプトを選択して実行するようになされていてもよい。

【0108】こうすることにより、例えば電話端末で接続するユーザ、ファクシミリ端末で接続するユーザ、PDA (personal digital assistant) などで接続するユーザそれぞれに対して、最適のサービスを提供することができるようになる。

【0109】2.5.12 サーバプログラム、データベースアクセス

本実施の形態のシステムのサーバで使用する応答スクリプトでは、インターネット2上の他のWWW情報を参照・利用することができる。インターネット2上の特定のwebページの内容を読み上げるスクリプトの一例としては、下記のようなスクリプトが考えられる。

【0110】

れている場合、下記のように、そのプログラムを利用したスクリプトは、接続してきたユーザに対し、現在時刻を知らせる。

【0111】

1の外部のインターネット2上に存在する情報を利用するように記述された部分を検知し、その情報をインター

ネット接続部3から直接、またはファイル転送制御部4を介して取り出すようになされている。

【0112】以上のように、本発明実施の形態に係るシステムは、既にインターネット上に存在する情報をも利用できるようになされており、応答スクリプトの作成者は自分が作成するスクリプトのもたらす情報をすべて自分で用意する必要はない。既にインターネット上に存在する情報を加工したり集計したりして作り出した情報を提供するようなサービスを行うこともできる。したがって、様々なサービスを提供するスクリプトをより多くの人々が作れる環境を提供することが可能である。

【0113】2.6 ダウンロードファイルのキャッシュの詳細

ファイル転送制御部4は、スクリプトインタプリタ実行部5の要求に応じてインターネット2経由でファイルをダウンロードするが、頻繁に使われるファイルは繰り返しダウンロードされることになる。同じファイルを何回もダウンロードするのは、インターネット2を通じた通信の場合には時間がかかる場合が多いので利用者の利便上に問題がある。

【0114】そこで、ダウンロードしたファイルをファイル一時記憶装置6に一時的に保存しておき、次に同じファイルをダウンロードするように指示された時に、そのファイル一時記憶装置6から取り出すようにすれば、同じファイルを再度ダウンロードするのに比べて時間がかからず効率が良い。このような処理は一般にファイルキャッシュと呼ばれる。

【0115】図14及び図15には、本実施の形態のシステムでファイルキャッシュを管理するためのファイル（リモートファイル管理ファイル、ローカルキャッシュ管理ファイル）の一例を示す。これらのファイルは、ファイル転送制御部4が作成／更新してファイル一時記憶装置6に置かれる。

【0116】図14のリモートファイル管理ファイルは、ダウンロードしたリモートファイルのURLとファイル一時記憶装置6内に格納したファイル名（ローカルファイル名）との対応関係などを記録しておく。なお、マスタファイルもリモートファイルと同様に扱ってもよい。ローカルファイル名は、リモートファイルをダウンロードしたときに、ファイル転送制御部4によって既にあるローカルファイル名と重ならないように生成されるようになされている。また、このファイルの項目として、リモートファイルが何時、作成／更新されたのかを記録するためのリモートファイル更新日時と、リモートファイルを何時、ダウンロードするかどうか調べたかを記録するための最終更新チェック日時などがある。この図14の例では、リモートファイルURL「http://test.jp/a.txt」は「a0121.txt」というファイル名でローカルファイルとして管理されていて、ダウンロードしたリモートファイルの作成／更新された日時は1997

年11月11日10時10分15秒で、最後にこのリモートファイルの更新日時を確認したのが1997年11月13日12時15分05秒であることを示している。

【0117】また、図15のローカルキャッシュ管理ファイルは、ローカルファイルのアクセスに関する情報を記録するためのものである。ローカルファイル毎に記録される項目として、ローカルファイルを最後にアクセスした日時である最終アクセス日時と現時点で該当するローカルファイルを使おうとしている（ダウンロード要求した）スクリプトの数であるアクセス中カウンタがある。このアクセス中カウンタは、その値が0のときはそのローカルファイルを使っていないことを示し、1以上の値のときは使われていることを示している。これによって、ローカルファイルを更新したり削除したりしてよいかどうかの判断が可能になる。この図15の例では、ローカルファイル「a0121.txt」が最後にアクセスされたのは1997年11月13日15時51分11秒であり、現在2つのスクリプトによって使われていることを示している。

【0118】2.6.1 キャッシュイン手順

図16にはリモートファイルをダウンロードするときにファイル転送制御部4がファイルキャッシュに関して行う手順を示す。これは、例えばスクリプトインタプリタ実行部5がスクリプトの記述に応じてリモートファイルのダウンロードをファイル転送制御部4に要求する場合などに実行される。

【0119】この図16において、先ずステップSP41では、リモートファイル管理ファイルのリモートファイルURLを検索キーにしてダウンロードしたいリモートファイルが登録されているか、つまり既にファイル一時記憶装置6にダウンロードされているか検索を開始する。ここで、ステップSP42では、リモートファイル検索が成功したか否かの判定を行い、検索が失敗したならば、つまり該当するリモートファイルが登録されていないならば、ステップSP47の処理に移ってダウンロードを行う。

【0120】一方、ステップSP42において、検索が成功、つまり該当リモートファイルが登録されているならば、再びダウンロードするかどうかを、以下の手順で調べて決める。

【0121】上記ステップSP42にてリモートファイルが登録されていると判定した場合において、ステップSP43では、リモートファイル管理ファイルの該当エントリに記録されている対応するローカルファイル名を取得し、ローカルキャッシュ管理ファイルから同じローカルファイル名のエントリを検索する。

【0122】次に、ステップSP44では、該当エントリのアクセス中カウンタを調べる。ここでは、エントリのアクセス中カウンタの値が1以上なら他にもローカルファイルをアクセスしているスクリプトなどがあると見

なしてダウンロードしないでステップSP51の処理に移る。一方、アクセス中カウンタの値が0ならば、ステップSP45の処理に移る。

【0123】ステップSP45では、インターネット接続部3を通してリモートのファイル記憶サーバ10などにリモートファイル自身の更新日時を問い合わせ、その結果を最新の更新日時として取得する。

【0124】次に、ステップSP46では、その最新の更新日時とリモートファイル管理ファイルに記録されているリモートファイル更新日時を比べ、同じならばリモートファイルが更新されておらずローカルファイルと同じものと見なしてダウンロードしないでステップSP50の処理に移る。

【0125】一方、ステップSP46での判定において、リモートファイルの更新日時が違っているならば、前回のダウンロードからリモートファイルが更新されていると見なして以下の手順でダウンロードを行う。

【0126】上記ステップSP46にてリモートファイルの更新日時が違っていると判定した場合において、ステップSP47では、キャッシュアウトチェックとしてファイル一時記憶装置6の記憶容量に余裕があるか否かを調べ、余裕が無いならばあまり使われそうにないローカルファイルを削除する。この動作の詳細は、この後の図17のフローチャートの解説の際に説明する。

【0127】次に、ステップSP48では、インターネット接続部3を通してファイル記憶サーバ10などに該当するリモートファイルの転送要求を行い、ファイル一時記憶装置6にダウンロードする。このときリモートファイル管理ファイルに登録されているファイルならば、既にある該当するローカルファイルを上書きし、登録されていないならば他のローカルファイルと重ならない名前を付けて新しいローカルファイルとして格納する。

【0128】次いで、ステップSP49では、リモートファイル管理ファイルとローカルキャッシュ管理ファイルの該当するエントリの内容を最新情報に更新する。ここでもし未登録のリモートファイルならば新しいエントリをそれぞれのファイルに作成する。リモートファイル更新日時とローカルファイルの最終アクセス日時は、現在の日時、アクセス中カウンタの値は0とする。

【0129】次のステップSP50では、リモートファイル管理ファイルの最終更新チェック日時を現在の日時に更新する。

【0130】その後、ステップSP51では、ローカルキャッシュ管理ファイルのアクセス中カウンタに1を加算（1インクリメント）し、該当するローカルファイルの利用を開始したことを記憶する。

【0131】最後に、ステップSP52では、ローカルキャッシュ管理ファイルの最終アクセス日時を現在の日時に更新し、このダウンロード処理を終わる。

【0132】ここで、もしこの処理がスクリプトインタ

ープリタ実行部5の要求によって行われたものならば、ダウンロードが終わったことをスクリプトインタプリタ実行部5に知らせる。

#### 【0133】2.6.2 キャッシュアウト手順

図17には、リモートファイルをダウンロードするときなどに、ファイル転送制御部4がファイルキャッシュから、使われる可能性が少ないローカルファイルを削除する手順（通常キャッシュアウトと呼ばれる）の一例を示す。この図17の例では、ファイル一時記憶装置6のローカルファイルを格納する領域に余裕ができるまで、最近アクセスされていないローカルファイルから順番に削除している。

【0134】この図17において、まずステップSP61では、ファイル一時記憶装置6の空き領域に余裕があるか否かを調べ、余裕があるならば処理を終了する。一方、もし余裕がないならばステップSP62の処理に進む。

【0135】ステップSP62では、ローカルキャッシュ管理ファイルの各エントリを最終アクセス日時の古い順に並べ、次のステップSP63では、最初のエントリ、つまり最終アクセス日時が一番古いものを取り出す。

【0136】次に、ステップSP64では、そのエントリのアクセス中カウンタの値が0かどうか調べ、0でないならば現在アクセス中なのでそのエントリを飛ばしてステップSP68の処理に進む。一方、アクセス中カウンタの値が0ならば、該当するローカルファイルを削除するため、まず、ステップSP65にてリモートファイル管理ファイルとローカルキャッシュ管理ファイルのそのローカルファイルに対応するエントリを削除する。

【0137】次に、ステップSP66では、そのローカルファイル自身を削除する。

【0138】次のステップSP67では、このローカルファイルの削除によってファイル一時記憶装置6の空き容量に余裕が出来たか否かを調べ、余裕が出来たならば処理を終了する。一方、まだ余裕がないならばステップSP68の処理に進む。

【0139】このステップSP68では、選択しているエントリが最後のエントリか否かを調べ、最後のエントリである場合には処理を終了し、最後のエントリでないならばステップSP69の処理に進む。

【0140】ステップSP69では、次のエントリを取得し、その後、ステップSP64の処理に戻って、再び削除できるローカルファイルを調べることを繰り返す。

【0141】図18には、ファイル転送制御部4がローカルファイルのアクセスを終了する手順を示す。これは、例えばスクリプトインタプリタ実行部5が1つのスクリプトの実行を終了するとき、そのスクリプトでダウンロードしたファイルについて利用が終わったことを知らせるために、ファイル転送制御部4に対して処理

の要求を行う。例えばもし図16の処理をスクリプトインタープリタ実行部5がスクリプトの記述に従って要求した場合、スクリプトインタープリタ実行部5がダウンロード要求したファイルを覚えておき、そのスクリプトを終了するときに、この覚えておいた全てのファイルについてその処理を要求する。同じファイルを複数回ダウンロード要求したならば、同じ回数だけこの処理を呼ぶことになる。

【0142】この図18において、先ずステップSP71では、ローカルキャッシュ管理ファイルから該当するローカルファイル名をキーにしてエントリを検索する。

【0143】次に、ステップSP72では、検索したエントリの最終アクセス日時を現在時刻に更新する。

【0144】次のステップSP73では、アクセス中カウンタの値から1を減算し、その後処理を終了する。これにより、基本的なファイルキャッシュの機能が可能になる。

#### 【0145】2.7 キャッシュ制御の拡張の詳細

前述したキャッシュ機能は、ダウンロードしたファイルを全てファイルキャッシュに入れて、ファイル一時記憶装置6の空き容量が足りなくなってきたら一番使われていないファイルから消していくという一般的なキャッシュ機能を、本実施の形態のシステムに適用したものである。また、ファイル記憶サーバ10にあるファイルが更新されているかどうかを毎回確認していた。

【0146】しかしながら、場合によってはファイル毎にキャッシュにおける扱いを指定できた方がよい場合がある。例えばマスタファイルなどでしばらく使われなくてもデフォルトのファイルとしてファイル一時記憶装置6上に残しておきたいファイルのような場合である。このようなファイルは、殆ど更新されないことが予めわかっているため、毎回更新チェックをする必要がないファイルである。またそれとは逆に、頻繁に更新されるために長期間キャッシュに残しておく必要が無いファイルもあり、これら様々な場合が想定される。前述したキャッシュ機能は、全てのファイルに対して同等の扱いをしていたが、以下の処理によってファイルそれぞれ若しくは一部のファイルについてのみ、キャッシュにおける扱いを特別なものにすることができる。

【0147】図19及び図20には、ファイルに対して個別にキャッシュにおける扱いを行うために使うファイルの一例を示す。

【0148】図19に示すサイトキャッシュ設定ファイルは、ファイル記憶サーバ10やマスタファイル記憶サーバ9のリモートファイルがあるディレクトリ（サイト）に、そのディレクトリの管理者が置く。このファイル自身もリモートファイルである。このファイルは、そのディレクトリ（サイト）上にあるリモートファイルそれぞれに関してファイル名とそのファイルの更新日時をチェックする頻度を示す更新日時チェック頻度、ファイ

ルをキャッシュ内にどれだけ置いておきたいかを示すファイルライフタイムが対応付けられている。この図19の例においては、例えばこのサイトキャッシュ設定ファイル自身のURLが「http://test.jp/d/sitec.def」とすると、同じサイトのファイル「a.wav」つまり「http://test.jp/d/a.wav」は更新される予定がないので更新日時チェックは不要で、キャッシュにできるだけ長く置いておくという設定である。また同様に、ファイル「b.scd」は1997年11月20日を過ぎたならば更新をチェックし、1997年11月30日を過ぎたならばキャッシュから削除してもよいという設定である。ファイル「sitec.def」は、更新チェックを前回のチェックから1時間以上過ぎたときに行い、キャッシュの扱いは通常通りにするという設定である。この設定のファイル名の指定には、ファイル毎の指定だけではなく、例えばワイルドカードや正規表現といった表現も可能であり、この図19の例のファイル「\*.txt」では例えば「abc.txt」や「foo.txt」などを含む。また、このファイル「\*.txt」は、頻繁に更新されるので、更新日時チェックをアクセス毎に行い、キャッシュはスクリプト実行中のみ有効という扱いになっている。

【0149】図20に示すキャッシュ設定対応ファイルは、リモートサイトからダウンロードしたサイトキャッシュ設定ファイルのローカルファイル名との対応を管理するためのものであり、リモートサイトURLとそのサイトのキャッシュ設定対応ファイルのローカルファイル名との対応を記録しておく。この図20の例においては、URLが「http://test.jp/d/」のリモートサイトからダウンロードしたキャッシュ設定対応ファイルはファイル「b0533.def」という名前前でファイル一時記憶装置6に格納されている。このファイルは、ファイル転送制御部4が作成／更新してファイル一時記憶装置6に置かれる。

#### 【0150】2.7.1 個別キャッシュインチェック

図21には、リモートファイルをダウンロードするときにファイル転送制御部4がファイルキャッシュに関して個別にロードチェックを行う手順を示す。この図21のフローチャートは、図16におけるステップSP44とステップSP45の間に挿入し、実行されることを想定している。

【0151】この図21において、先ずステップSP81では、キャッシュ設定対応ファイルからリモートサイトのURLを検索する。

【0152】次のステップSP82では、ステップSP81においてダウンロード要求されたリモートファイルのURLに対応するリモートサイトのURL（リモートファイルURLが「http://www.test/d/a.wav」とすると、リモートサイトのURLは「http://www.test/d/」）がキャッシュ設定対応ファイルに登録されているかどうかを検索する。このステップSP82において、登

録されている場合はステップSP86の処理に移り、登録されていない場合はステップSP83の処理に移る。

【0153】ステップSP83では、該当するキャッシュ設定対応ファイルをダウンロードする。

【0154】次のステップSP84では、ダウンロードが成功したか否かを調べ、ここでもしキャッシュ設定対応ファイルがダウンロードできなかったならば、ステップSP91の処理に移り、ここから図16のステップSP45の処理に続くことになる。

【0155】一方、ステップSP84にてキャッシュ設定対応ファイルがダウンロードできた場合は、ステップSP85の処理に移る。このステップSP85では、キャッシュ設定対応ファイルに、そのリモートサイトURLとローカルファイル名を登録する。

【0156】次に、ステップSP86では、該当するサイトキャッシュ設定ファイルの各エントリのファイル名の中に、ダウンロード要求されたリモートファイルの名前にマッチするものがあるかどうか検索する。このとき、ファイル名にワイルドカードや正規表現などがなされていた場合はそのように検索する。

【0157】次のステップSP87では、その検索の結果、該当するリモートファイルのエントリが無かったならば、ステップSP91を介して図16のステップSP45の処理に進み、エントリが見つかったらステップSP88の処理に進む。

【0158】ステップSP88では、該当するエントリの更新日時チェック頻度を取得する。

【0159】そして、ステップSP89では、更新日時チェック頻度によって更新日付チェックが必要かどうかを判断し、必要ならばステップSP91を介して図16のステップSP45の処理に進み、もし更新日付が不要ならばステップSP92を介して図16のステップSP51の処理に進む。

【0160】このようにして、リモートファイル個別に更新日時のチェックが必要かどうかを調べ、処理する。

【0161】2.7.2 個別キャッシュアウトチェック  
図22にはトラストサイト設定ファイルを示す。このトラストサイト設定ファイルは、信用できるリモートサイトを登録しておくファイルで、管理者が作成してマスタファイル記憶サーバ9に格納しておき、これをシステム立ち上げ時などにファイル転送制御部4がファイル一時記憶装置6にダウンロードするようになっている。このファイルは、特定のリモートサイトのサイトキャッシュ設定ファイルなどの記述をそのまま信用するかどうか判断するのに使われ、ファイル転送制御部4やスクリプトインタープリタ実行部5が参照する。この図22の例では「http://master.jp/a/」と「http://test.jp/d/」というURLで表されるリモートサイトが、信用できるサイトとして登録されている。

【0162】図23には、使われないファイルをファイ

ルキャッシュから削除する場合などにファイル毎に予め設定された寿命チェックを行う手順を示す。このフローチャートは、図17におけるステップSP64とステップSP65の間に挿入し、実行されることを想定している。

【0163】この図23において、先ずステップSP101では、キャッシュから削除しようとしているローカルファイルのリモートファイルURLをリモートファイル管理ファイルから取得する。

【0164】次のステップSP102では、このリモートファイルURLに対応するリモートサイトURLがキャッシュ設定対応ファイルに登録されているかどうか検索する。

【0165】次に、ステップSP103では、リモートサイトURLの検索が成功したか否かを調べ、登録されていなかったらステップSP110を介して図17のステップSP65の処理に移ってローカルファイルを削除するが、登録されていれば次のステップSP104の処理に進む。

【0166】ステップSP104では、キャッシュ設定対応ファイルから該当するサイトキャッシュ設定ファイルを特定し、そのファイルからリモートファイル名を検索する。

【0167】次のステップSP105では、ファイル名の検索が終了したか否かを調べ、この検索によってリモートファイルのエントリが見つからなかったらステップSP110を介して図17のステップSP65の処理に移り、見つかったらステップSP106の処理に進む。

【0168】ステップSP106では、該当するエントリのファイルライフタイムを取得する。

【0169】次に、ステップSP107では、ファイルライフタイムの設定が永久保存等、通常の設定でなく且つまだ削除しない設定となっているか否かを調べ、まだ削除しない設定となっている場合は、ステップSP108の処理に進むが、そうでなければステップSP110を介して図17のステップSP65の処理に移る。

【0170】ステップSP108では、サイトキャッシュ設定ファイルの内容を信用してもよいか調べるためにトラストサイト設定ファイルから該当するリモートサイトを検索する。

【0171】次のステップSP109では、リモートサイトURLの検索が成功したかどうか調べ、もしその検索の結果、リモートサイトが見つからなかったらステップSP110を介して図17のステップSP65の処理に移り、見つかったらステップSP111を介して図17のステップSP68の処理に移ってファイルの削除をしない。

【0172】ステップSP110では、図17のステップSP65の処理に進むが、もし削除するファイルがサ

イトキャッシュ設定ファイルであるならば、キャッシュ設定対応ファイルから該当するエントリを削除する処理を追加する。

【0173】このようにして、ファイル毎に予め設定されていたファイルの寿命を考慮してキャッシュから削除するタイミングを調整することができる。

【0174】図24には、ローカルファイルアクセス終了時などに、ファイル転送制御部4が必要なくなったファイルか調べて削除する手順を示す。この図24のフローチャートは、図18のステップSP73の後に挿入されて実行されることを想定している。

【0175】この図24において、先ずステップSP121では、アクセス終了したファイルに関してローカルキャッシュ管理ファイルの中のアクセス中カウンタが0になったかどうか調べ、0でなければ処理を終了する。一方、もしアクセス中カウンタが0になったら、ステップSP122の処理に進む。

【0176】ステップSP122とそれに続くステップSP123では、図23のステップSP101及びステップSP102と同じ処理を行う。すなわち、キャッシュ設定対応ファイルの該当するリモートサイトのエントリを検索する。

【0177】次のステップSP124では、リモートサイトURLの検索が成功したかどうか調べ、検索できずにエントリが見つからなかったら処理を終了し、見つかったら次のステップSP125の処理に進む。

【0178】ステップSP125では、図23のステップSP104と同じ処理を行う。つまり、サイトキャッシュ設定ファイルから該当するリモートファイルのエントリを検索する。

【0179】次のステップSP126では、ファイル名の検索が成功したかどうか調べ、もしファイル名が無ければ処理を終了し、見つければステップSP127の処理に移る。

【0180】ステップSP127では、該当エントリのファイルライフタイムを取得する。

【0181】次に、ステップSP128では、ファイルライフタイムの設定によってそのファイルの寿命が来ている、つまり1つのスクリプトがアクセス中しかキャッシュに残さない設定やキャッシュに残す日付の設定が既に過ぎているかどうか調べ、削除する設定になっている場合には、次のステップSP129の処理に進み、そうでなければ処理を終了する。

【0182】ステップSP129とそれに続くステップSP130では、図17のステップSP65及びステップSP66と同様に、各管理ファイルからエントリを削除し、ローカルファイル自身を削除する。その後は処理を終了する。

【0183】このようにしてスクリプト終了時などで随時いらなくなったファイルを個別にファイルキャッシュ

から削除していくことが可能になる。

【0184】2.8 ダウンロードファイルの先読みの詳細

ファイル転送制御部4は、スクリプトインタプリタ実行部5の要求に応じてリモートファイルや、マスタファイル9またはサーバプログラムの実行結果をダウンロードするが、インターネット2を使った通信を行うので、LAN (Local Area Network) 内で構成されたシステムよりもダウンロードに時間がかかることが多い。スクリプトファイル内のファイルのダウンロードを指示する部分で毎回ダウンロードを行っていると、電話端末15などを使ってサーバ1にアクセスしている利用者をその都度待たせることになるので利便上問題がある。

【0185】そこで、本実施の形態では、ファイルのダウンロードをスクリプトで明示的に指示される部分よりも前から、他の処理と平行して行うようにすることで、上記の問題を略々解消するようにしている。

【0186】例えば、電話端末15を使ってサーバ1にアクセスする利用者に対してスクリプトに従って音声再生などを行っている間に、その後にダウンロードする可能性があるファイルを予めダウンロードする。

【0187】図25には、スクリプトを実行するときにファイルを先読みする処理をスクリプトインタプリタ実行部5が行う手順を示す。

【0188】この図25のフローチャートにおいて、先ずステップSP141では、スクリプトを実行する前に、そのスクリプト中でダウンロードされる可能性があるファイル名を取得する。

【0189】次のステップSP142では、取得したファイル名をファイル転送制御部4に送って、それらのファイルのダウンロードを開始してもらう。この後から実際にスクリプトの手順に従って実行を行う。

【0190】ステップSP143では、スクリプトの最初のコマンドを読む。

【0191】次のステップSP144では、そのコマンドがダウンロードのコマンドか調べ、ダウンロードするコマンドならばステップSP145の処理に進み、それ以外のコマンドならステップSP147の処理に移る。

【0192】ステップSP145では、ダウンロードするファイル名が既にステップSP142でダウンロードを開始したファイルかどうか調べ、そのファイルであるならばステップSP150の処理に移り、そのファイルでないならばステップSP146の処理に進む。

【0193】ステップSP146では、そのファイル名をファイル転送制御部4に送ってダウンロードを開始してもらう。

【0194】一方、ステップSP147においては、ダウンロードするコマンド以外のコマンドのうち、音声ファイルの再生などダウンロードしたファイルを使うコマンドかどうか調べ、違えばステップSP149の処

理に移ってそれぞれのコマンドを実行する。これに対して、そのコマンドであるならばステップSP148の処理に移る。

【0195】ステップSP148では、ファイル一時記憶装置6にそのファイルがダウンロードされたか調べて、未だダウンロードされていないならばダウンロードされるまで待つ。

【0196】ステップSP148にてダウンロードがなされたならば、ステップSP149の処理に移ってそのコマンドを実行する。

【0197】最後に、ステップSP150では、スクリプトが終了したかどうか調べ、まだ終了していないならばステップSP151の処理に移って次のコマンドをスクリプトから読んだ後にステップSP144の処理まで戻って同じ処理を繰り返す。

【0198】一方、ステップSP150にてスクリプトが終了したならば、そのスクリプトの実行を終わる。

【0199】次に、図26と図27を使って実際の動作の例を説明する。

【0200】図26には、例として2つの音声ファイルを順番にダウンロードして再生するスクリプト自身の記述手順を示す。

【0201】この図26において、スクリプトの内容としては、先ずステップSP161で音声ファイルAをファイル記憶サーバ10からダウンロードする。

【0202】次のステップSP162では、ダウンロードした音声ファイルAを再生する。

【0203】同様にして、ステップSP163では、音声ファイルBをダウンロードし、ステップSP164では、ダウンロードした音声ファイルBを再生する。

【0204】図27には、図26のスクリプトを実際にスクリプトインタプリタ実行部5が行う手順を示す。

【0205】この図27において、先ず、ステップSP171では、図26のスクリプトからダウンロードする可能性があるファイルのファイル名（音声ファイルAと音声ファイルB）を抽出する。これは、図25のステップSP141の処理に対応する。

【0206】次に、ステップSP172では、抽出したファイル名（音声ファイルAと音声ファイルB）をファイル転送制御部4に送ってダウンロードを要求する。これは、図25のステップSP142の処理に対応する。

【0207】次に、ステップSP173では、スクリプトの先頭コマンドである図26のステップSP161を読み、ダウンロード要求コマンドでしかも既にダウンロード要求済なので、次のコマンドに進む。これは、図25のステップSP143からステップSP145の処理、ステップSP150からステップSP151の処理に対応する。また、図26のステップSP162は、ダウンロード要求済ファイル（音声ファイルA）を利用するコマンドなので、ファイル一時記憶装置6にそのファ

イルがダウンロード完了するまで待つ。これは、図25のステップSP144、ステップSP147からステップSP148の処理に対応する。

【0208】そして、ステップSP174では、ダウンロードが完了した音声ファイルAを電話回線網接続部7の音声録音・再生部7Bにて再生してもらう。これは図25のステップSP149の処理に対応する。

【0209】ステップSP175とステップSP176も、ステップSP173とステップSP174と同様にして音声ファイルBを再生し、スクリプトの実行を終了する。

【0210】

【発明の効果】以上の説明で明らかなように、本発明においては、コンピュータネットワークを介して取得し、記憶した情報を個別に制御可能となすことにより、インターネット等を介して情報を取得する場合に使用されるキャッシュの機能をユーザに要求に応じて設定可能である。

【図面の簡単な説明】

【図1】本発明の情報処理装置が適用されるシステムの基本的な構造を示すブロック図である。

【図2】本発明実施の形態のシステムの全体構成を示すブロック図である。

【図3】スクリプト管理ファイルの構造を示す図である。

【図4】応答スクリプトと実行手順を示すフローチャートである。

【図5】応答スクリプトの登録手順を示すフローチャートである。

【図6】画面上の登録フォームの一例を示す図である。

【図7】画面上の登録結果通知の一例を示す図である。

【図8】サービス利用時の手順を示すフローチャートである。

【図9】複数サービスが混在した例の説明に用いる図である。

【図10】全体構成図のうち、音声合成に関する部分のみを抜き出して示すブロック図である。

【図11】データファイルをサーバの外部に置いた例の説明に用いるブロック図である。

【図12】応答スクリプトをサーバの外部に置いた例の説明に用いるブロック図である。

【図13】ファイル転送制御部の位置関係の説明に用いるブロック図である。

【図14】リモートファイル管理ファイルの一例を示す図である。

【図15】ローカルキャッシュ管理ファイルの一例を示す図である。

【図16】ファイル転送制御部のファイルキャッシュに関して行う処理手順を示すフローチャートである。

【図17】ファイル転送制御部のキャッシュアウト処理

手順を示すフローチャートである。

【図 18】ファイル転送制御部のファイルアクセス終了時の処理手順を示すフローチャートである。

【図 19】サイトキャッシュ設定ファイルの一例を示す図である。

【図 20】キャッシュ設定対応ファイルの一例を示す図である。

【図 21】ファイル転送制御部の個別ロードチェック処理手順を示すフローチャートである。

【図 22】トラストサイト設定ファイルの一例を示す図である。

【図 23】ファイル毎の寿命チェック処理手順を示すフローチャートである。

【図 24】ファイルアクセス終了後などでの不要になったファイルの削除処理手順を示すフローチャートである。

【図 25】ダウンロードファイルの先読みでスクリプトインタプリタが行う処理手順を示すフローチャートである。

【図 26】2つの音声ファイルを再生するスクリプトの

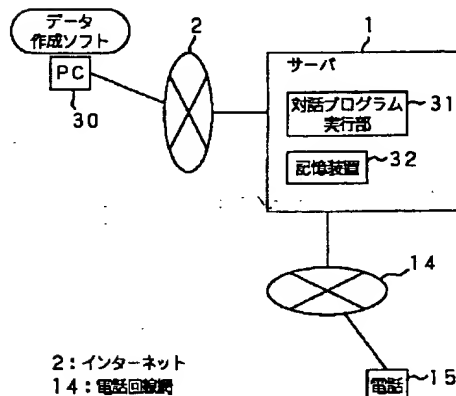
例の処理手順を示すフローチャートである。

【図 27】図 26 のスクリプトの実行手順の例を示すフローチャートである。

【符号の説明】

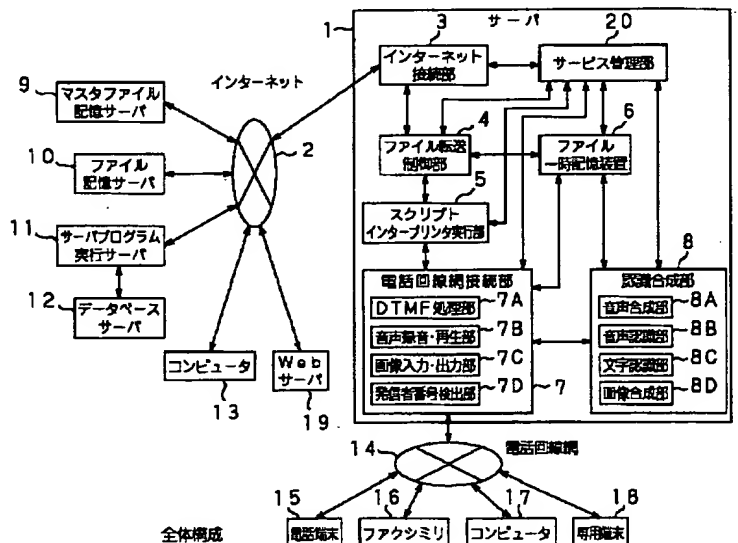
1 サーバ、2 インターネット、3 インターネット接続部、4 ファイル転送制御部、5 スクリプトインタプリタ実行部、6 ファイル一時記憶装置、7 電話回線網接続部、7A DTMF 処理部、7B 音声録音・再生部、7C 画像入力・出力部、7D 発信者番号検出部、8 認識合成部、8A 音声合成部、8B 音声認識部、8C 文字認識部、8D 画像合成部、9 マスタファイル記憶サーバ、10 ファイル記憶サーバ、11 サーバプログラム実行サーバ、12 データベースサーバ、13 コンピュータ (PC)、14 電話回線網、15 電話端末、16 ファクシミリ端末、17 コンピュータ、18 専用端末、19 webサーバ、20 サービス管理部、30 PC、31 対話プログラム実行部、32 記憶装置

【図 1】



基本的構成

【図 2】



全体構成

【図 3】

対応サービスコード	スクリプトファイル名
3344	file0001.sd
3345	file0002.sd
5348	http://banana/users/yamada/sd/test.sd
...	...

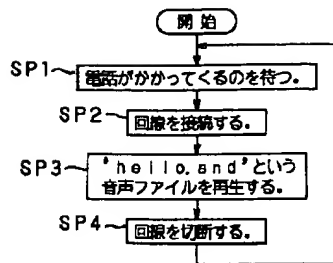
スクリプト管理ファイルの構造

【図 14】

リモートファイルURL	ローカルファイル名	リモートファイル更新日時	最終更新チェック日時
http://test.fw.txt	a0121.txt	1997/1/11 10:10:15	1997/1/13 12:15:05
...	...	...	...

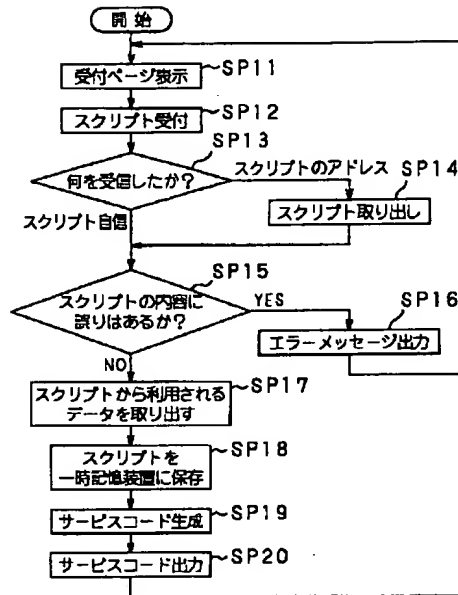
リモートファイル管理ファイル

【図4】



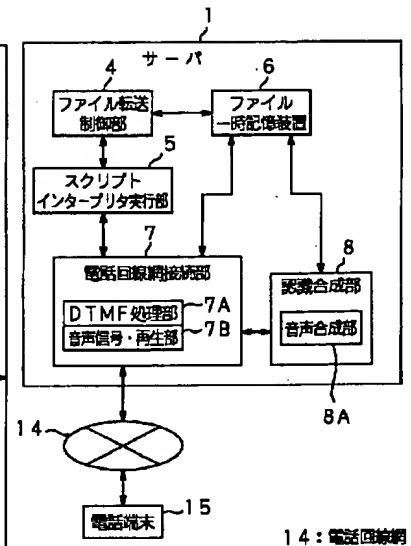
応答スクリプトの実行手順

【図5】



応答スクリプトの登録手順

【図10】



音声合成関連部分

【図6】

図6は「応答スクリプト登録フォーム」のスクリーンショットである。フォームには「あなたの名前:」と「ファイル:」の入力欄があり、「OK/RESET」ボタンが配置されている。ブラウザのアドレスバーには「http://test.ip/d/」が表示されている。

登録フォーム

【図7】

図7は「登録結果」のスクリーンショットである。画面には「あなたの応答スクリプトファイルは正しく登録されました。この応答スクリプトのアクセス番号は3487です。」というメッセージが表示されている。ブラウザのアドレスバーには「http://test.ip/d/」が表示されている。

登録結果通知

【図15】

ローカルファイル名	更新アクセス日時	アクセス中カウンタ
a0121.txt	1997/1/13.15:51:11	2
...	...	...

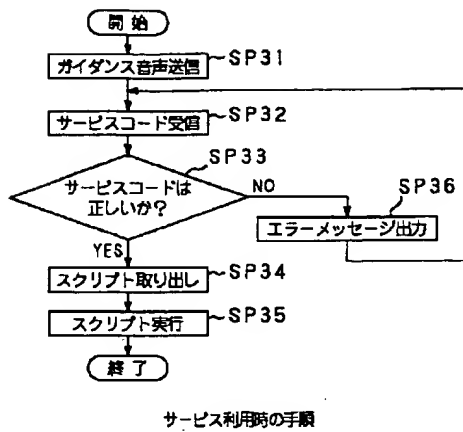
ローカルキャッシュ管理ファイル

【図20】

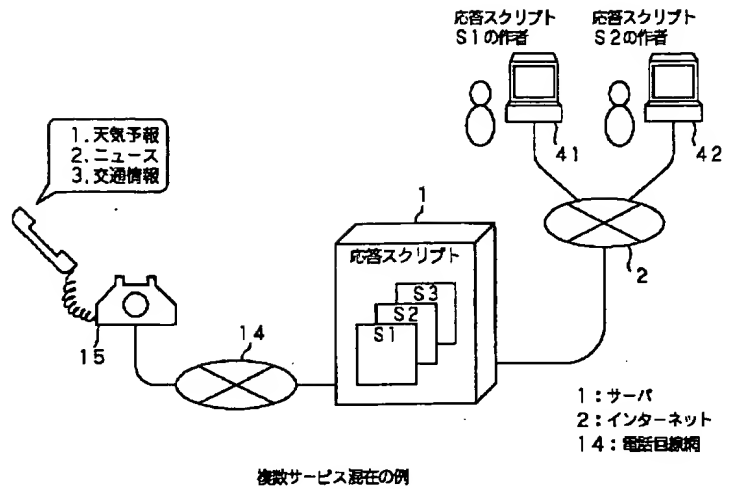
リモートサイトURL	ローカルファイル名
http://test.ip/d/	b0533.def
...	...

キャッシュ設定対応ファイル

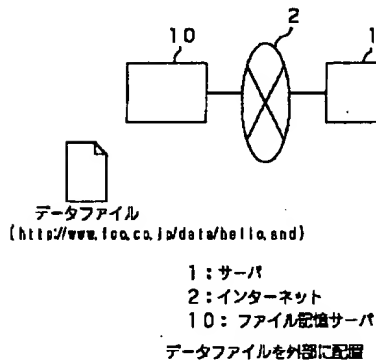
【図 8】



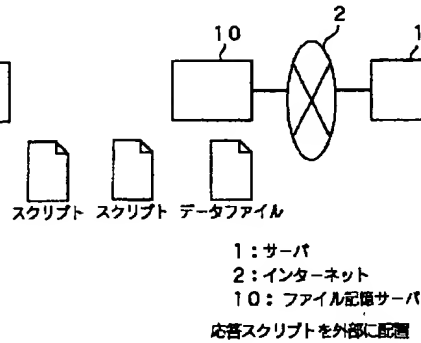
【図 9】



【図 11】



【図 12】

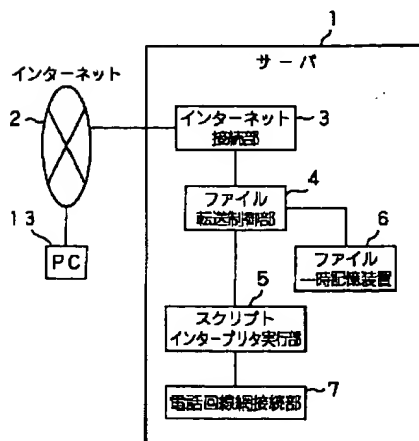


【図 22】

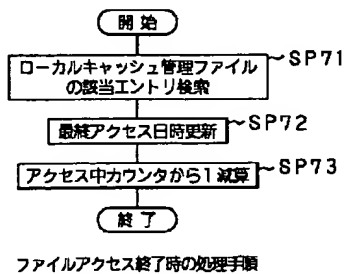
リモートサイトURL
<a href="http://master.jp/a/">http://master.jp/a/</a>
<a href="http://test.jp/d/">http://test.jp/d/</a>
...

トラストサイト設定ファイル

【図 13】



【図 18】

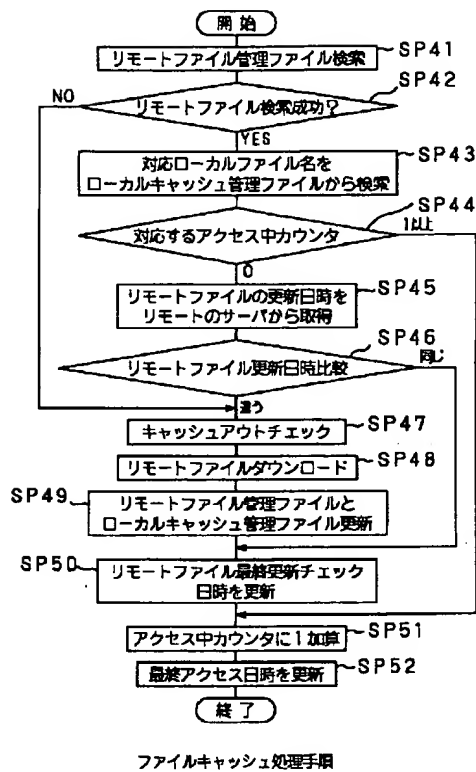


【図 19】

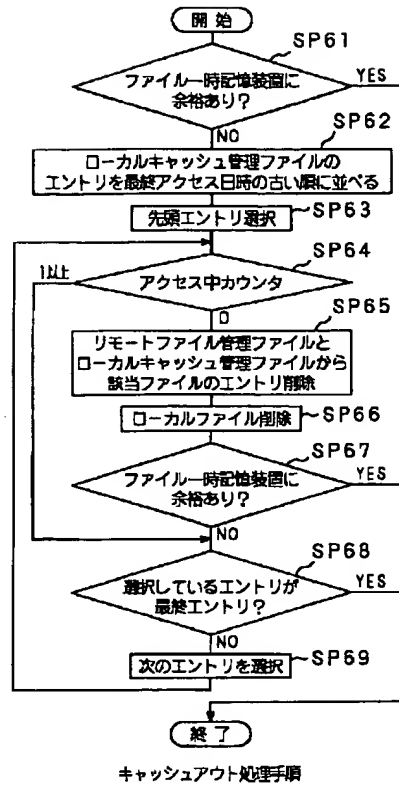
ファイル名	更新日時チェック頻度	ファイルライフタイム
a.wav	チェック不要	永久保存
b.acd	1997/11/20	1997/11/30
sitec.def	1時間後	通常
*.txt	アクセス毎	スクリプト実行中

サイトキャッシュ設定ファイル

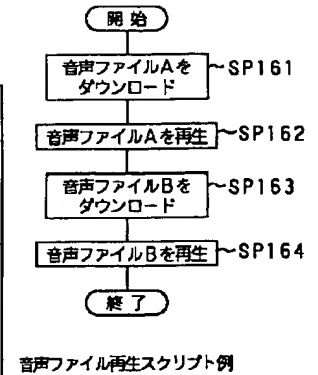
【図16】



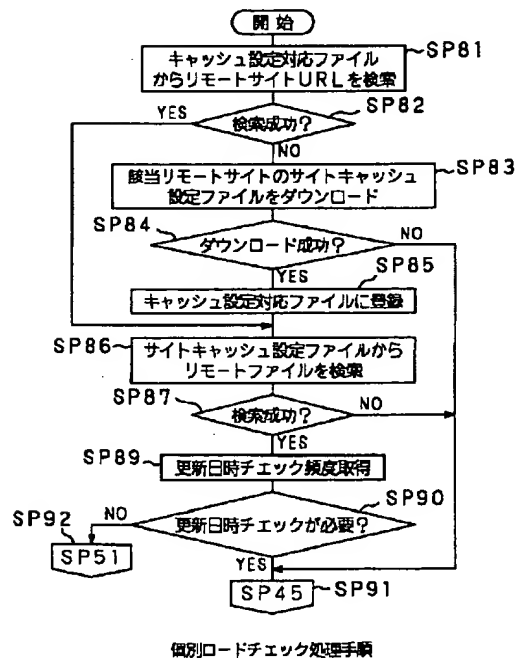
【図17】



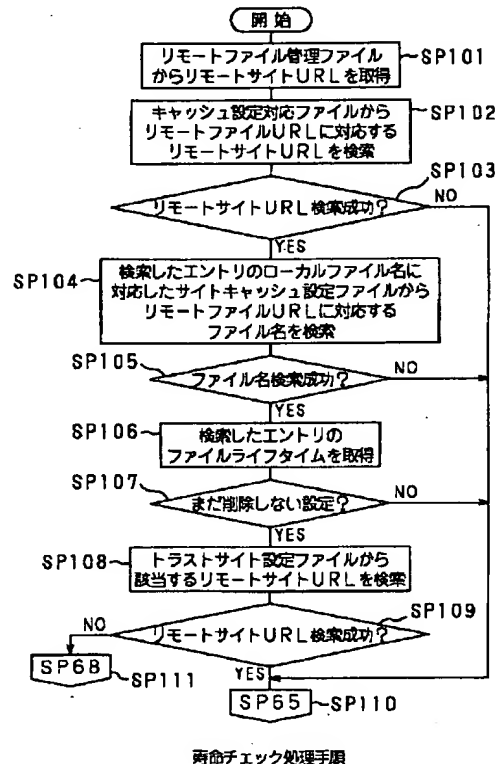
【図26】



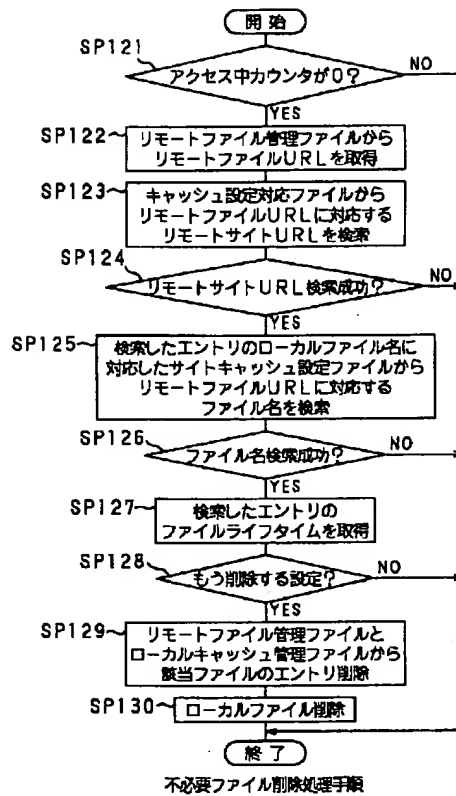
【図21】



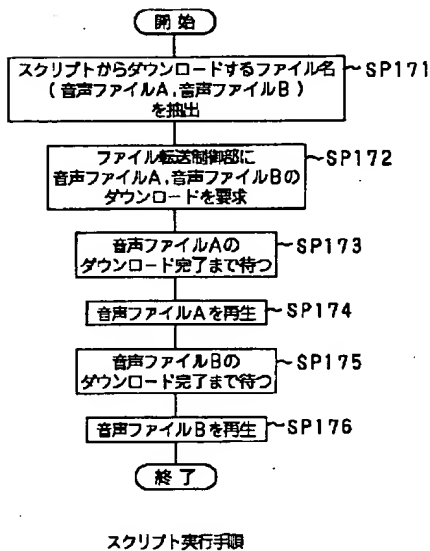
【図23】



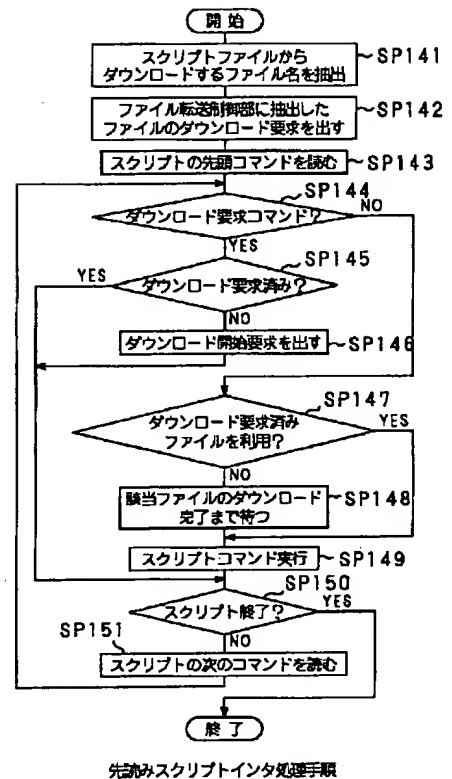
【図24】



【図27】



【図25】



フロントページの続き

(72)発明者 小野木 渡  
東京都品川区北品川6丁目7番35号 ソニ  
ー株式会社内

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**